Cross validation

Garyfallos Konstantinoudis

Mar 7, 2025

Cross validation

Tools that involve repeatedly drawing samples from a training set and refitting a model on each sample. In each draw we obtain more information about the fitted model.

Aims

- To evaluate prediction rules and compare different models with respect to their predictive performance.
- To fix open parameters and set model complexity, e.g. λ the regularization parameter in regularized regression.

CV approaches

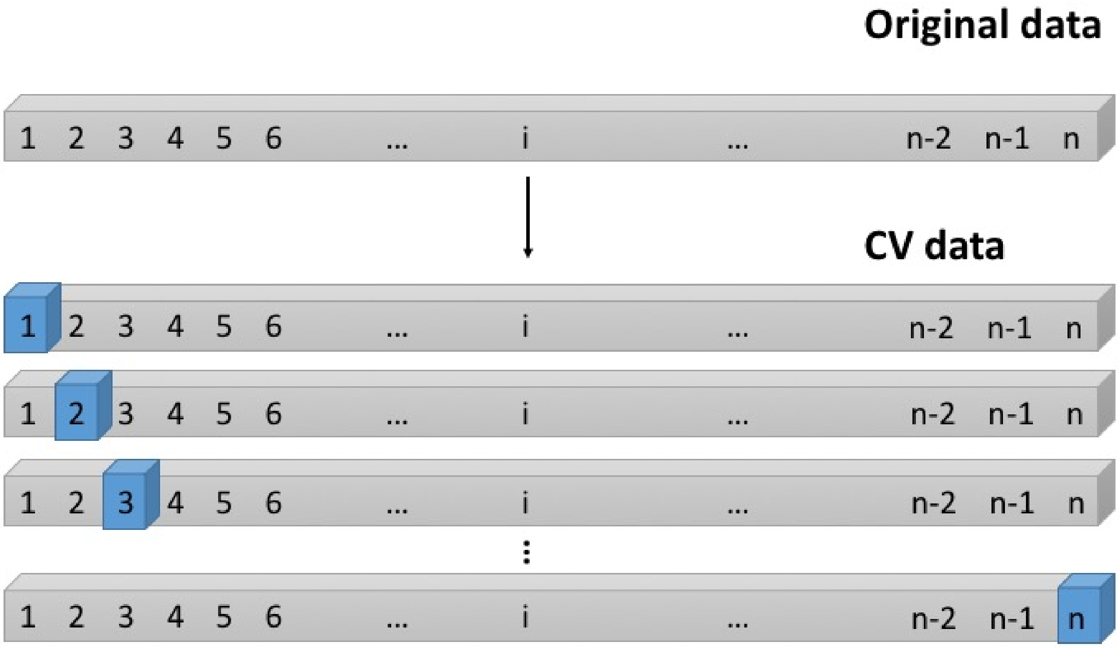
- Exhaustive cross-validation
  - Leave-one-out cross-validation
  - Leave-p-out cross-validation
- Non-exhaustive cross-validation
  - k-fold cross-validation
  - Repeated random sub-sampling validation

Leave-one-out cross-validation (LOOCV)

- Split the data containing n observations into
  - Training data of size $n-1$
  - Test data of size 1
- In each split, we leave out **one** observation.
- We fit the prediction rule $\hat{f}(x)$ on the training data without observation i.
- We evaluate the MSE_i of $\hat{f}(x_i)$ on the single observation i.
- Repeat n-times for $i \in 1, ..., n$.
- Overall mean CV test error is defined as

$$CV_{\{n\}} = \frac{1}{n}\sum_{i=1}^{n} MSE_i$$

Leave-one-out cross-validation (LOOCV)


Original data
1 2 3 4 5 6 ... i ... n-2 n-1 n
CV data
1 2 3 4 5 6 ... i ... n-2 n-1 n
1 2 3 4 5 6 ... i ... n-2 n-1 n
1 2 3 4 5 6 ... i ... n-2 n-1 n
⋮
1 2 3 4 5 6 ... i ... n-2 n-1 n

Leave-p-out cross-validation (LOOCV)

- Split the data containing n observations into
  - Training data of size $n-p$
  - Test data of size p
- In each split, we leave out p observations.
- We fit the prediction rule $\hat{f}(x)$ on the training data without the p observations.
- We evaluate the MSE_i of $\hat{f}(x_i)$ on the test data $i \in p$.
- Repeat for all possible combinations $comb = \binom{n}{p}$ of how to select p elements from a set of n.
- Overall mean CV test error is defined as:

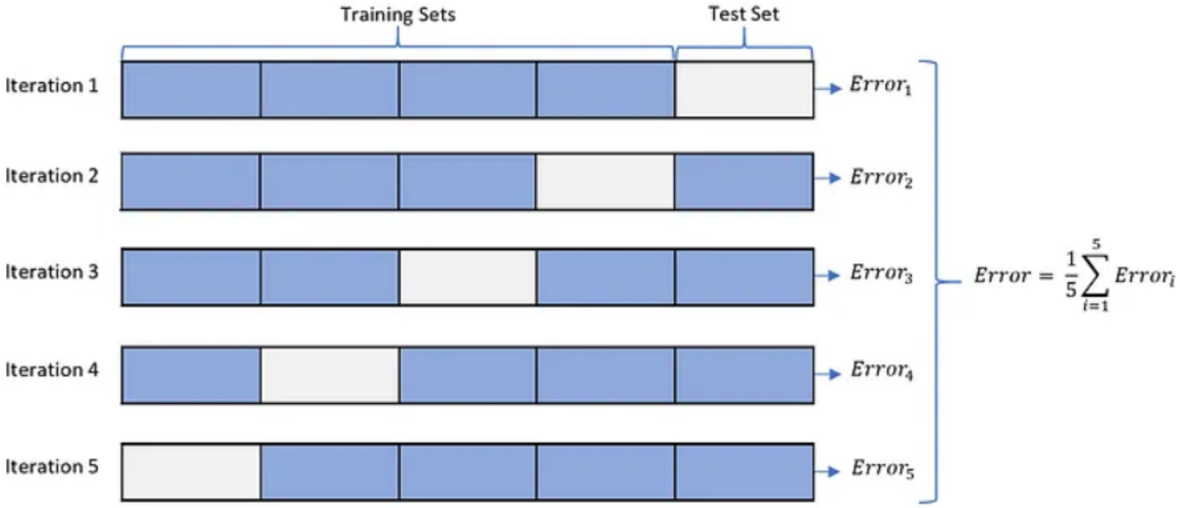
$$CV_{\{comb\}} = \frac{1}{comb} \sum_{i=1}^{comb} MSE_i$$

k-fold cross-validation

- With k-fold CV, we divide the data set into k different subsets, each of the same length.
- Recommended are $k=5$ or $k=10$.
- We fit the prediction rule $\hat{f}(x)$ on the training data including $k-1$ subsets.
- We evaluate the MSE_g of $\hat{f}(x_g)$ on all bservations g in subset k.
- Repeat k-times for $g \in 1, ..., k$.
- Overall CV test error rate is defined as

$$CV_{k-fold} = \frac{1}{k} \sum_{g=1}^{k} MSE_g$$

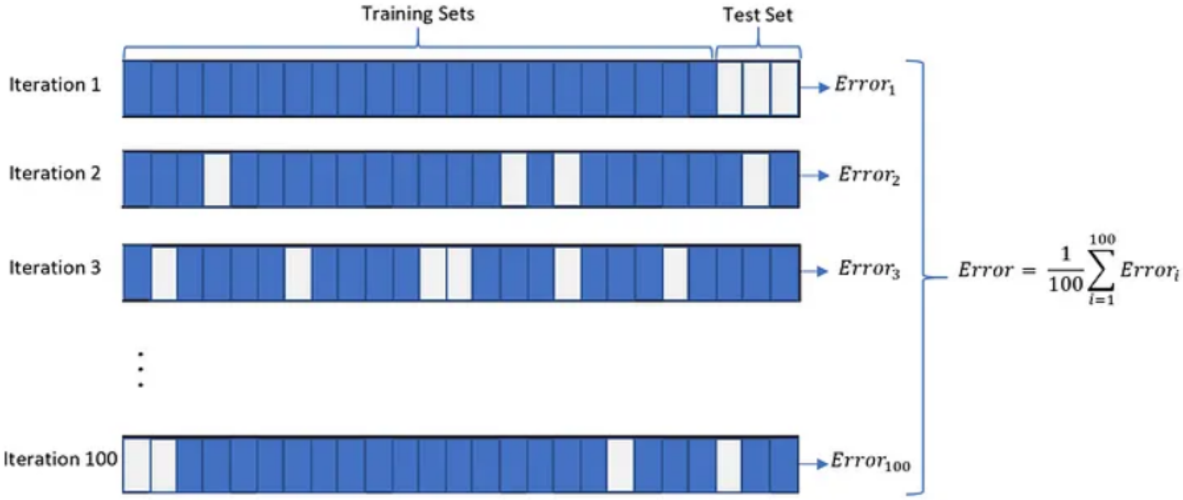
Leave-p-out cross-validation (LOOCV)



Repeated random sub-sampling validation

- Also known as Monte Carlo CV.
- Randomly splits the dataset into training and test data.
- Advantage: The proportion of the training/test split is not dependent on the folds.
- No guarantee that the samples are evenly distributed among training and test data, e.g. some samples might only ever be in the training data and never used to test the prediction.

MC cross validation



Cross-validation in R: crossval

1. Write a prediction function

```
predfun.lm = function(train.x, train.y, test.x, test.y){
  lm.fit = lm(train.y~., data=train.x)
  ynew = predict(lm.fit, test.x)

  # compute squared error risk (MSE)
  out = mean( (ynew - test.y)^2 )
  return(out)
}
```

Cross-validation in R: crossval

2. Load the `crossval` package and perform the CV for model `x` (all data) and `x1` (first 6 columns).

```
library(lars)
library(crossval)

data(diabetes)
```

```
x <- diabetes$x
y <- diabetes$y
x1 <- x[,1:6]

set.seed(11)
cv.out = crossval(predfun.lm, x, y, K = 5, verbose = FALSE)
cv.out1 = crossval(predfun.lm, x1, y, K = 5, verbose = FALSE)
```

Cross-validation in R: crossval

3. Evaluate the CV test error rate for `x`, and `x1`:

```
cv.compare = rbind(c(cv.out$stat, cv.out$stat.se),
                   c(cv.out1$stat, cv.out1$stat.se))
colnames(cv.compare) <- c("CV", "se")
rownames(cv.compare) <- c("x", "x1")
cv.compare
```

```
         CV       se
x  3013.887 37.90139
x1 3661.270 43.49090
```

4. Model x has lower CV test error than x1